



⑮ **BUNDESREPUBLIK
DEUTSCHLAND**



**DEUTSCHES
PATENT- UND
MARKENAMT**

⑫ **Offenlegungsschrift**
⑩ **DE 101 50 391 A 1**

⑤① Int. Cl.⁷:
G 06 F 17/50

⑳ Aktenzeichen: 101 50 391.1
㉔ Anmeldetag: 8. 10. 2001
㉕ Offenlegungstag: 8. 5. 2002

DE 101 50 391 A 1

③① Unionspriorität:
680751 06. 10. 2000 US

⑦① Anmelder:
CoCreate Software GmbH & Co. KG, 71065
Sindelfingen, DE

⑦④ Vertreter:
Hössle & Kudlek, 70184 Stuttgart

⑦② Erfinder:
Matheson, Dan, Fort Collins, Col., US

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

⑤④ Objektmodell und Verfahren zum Erfassen von Informationen

⑤⑦ Objektmodell zum Erfassen von Informationen, die produktinnovationsbezogene Daten betreffen, mit einer Produktideeschnittstelle zum Erfassen einer Idee für ein Produkt in einem Produktideeobjekt und einer Gestaltungsalternativenschnittstelle zum Erfassen einer Vielzahl von Gestaltungsalternativen für das Produkt in einer Vielzahl von jeweiligen Gestaltungsalternative-Objekten.

DE 101 50 391 A 1

Beschreibung

[0001] Die vorliegende Erfindung betrifft allgemein die Informationsverwaltung und im besonderen ein System bzw. Objektmodell und ein Verfahren zum Erfassen und Speichern von und Zugreifen auf Informationen, die während der Innovation eines Produktes in einer beständigen und werkzeugneutralen Form verwendet werden.

[0002] Produktentwicklung wird heutzutage von computerbasierten Anwendungen unterstützt, u. a. durch Textverarbeitungs- und Grafikwerkzeuge, Abwicklungswerkzeuge (Terminkoordinierung/Ablaufsteuerung) und Produktdatenverwaltungswerkzeuge. Der typische Produktentwicklungszyklus beginnt mit einer Idee für ein Produkt oder einer Verbesserung für ein Produkt, das ein Bedürfnis in der Industrie betrifft oder eine Lösung für ein Problem darstellt. Ausgehend von der Produktidee können unterschiedliche Entwürfe (Alternativgestaltungen) untersucht werden und schließlich wird ein Entwurf bzw. eine Gestaltung ausgewählt, entwickelt/umgesetzt und implementiert. Während der Anfangsphasen des Produktentwicklungszyklus werden Textverarbeitungs-, Grafik- und Ablaufwerkzeuge oft verwendet, um Informationen wie Marketinganalysen, projektierte Entwicklungsabläufe und Beschreibungen und Begründungen, die bestimmten ausgewählten Entwürfen zugrunde liegen, zu erfassen. Während der Entwurfs- bzw. Gestaltungsphase werden die Gestaltung betreffende Informationen, wie die Gestaltungsspezifikationen und 3-D-Modell-daten, üblicherweise unter Verwendung eines CAD-Werkzeugs erfaßt. Während der Herstellung des Produkts werden Informationen zur Teileverfolgung üblicherweise unter Verwendung eines PDM-Werkzeugs (PDM: Product Data Management, Produktdatenverwaltung) erfaßt. Zur Veranschaulichung soll davon ausgegangen werden, daß ein Entwickler/Gestalter eine Idee für ein neues Produkt hat. Der Entwickler ist sich dabei mehrerer Anforderungen bewußt, die das Produkt erfüllen muß und hat einige Lösungsideen. Der Entwickler muß mehrere unterschiedliche Werkzeuge benutzen, um Darstellungen unterschiedlicher Teile der Lösung zu erzeugen. Beispielsweise benutzt der Entwickler Microsoft Excel zum Erzeugen einer Kostenanalyse, Corel zur graphischen Darstellung, SolidDesigner für ein erstes Raumbudget und Work-Manager von CoCreate zur Erzeugung einer ersten Funktionsorganisation.

[0003] Während es klar ist, daß unterschiedliche computerbasierte Werkzeuge die Erfassung von Informationen und das Verfolgen des Fortschritts eines Produktes unterstützen, weist der Stand der Technik einige Nachteile auf. Erstens existiert derzeit kein Werkzeug, das konkret zur Erfassung und Verfolgung von Ideen und Entscheidungen zu diesen Ideen während der Anfangsphasen der Produktentwicklung dient. Die Untersuchung von Ideen ist häufig eine Situation, in der viel Empirie bzw. Regula falsi (trial and error) betrieben wird. Um eine derartige Untersuchung erfolgreich verfolgen zu können, ist es notwendig, viele nur teilweise vollständigen Informationsstrukturen, die Entscheidungen, einen Untersuchungsweg weiterzuverfolgen oder aufzugeben, und die Erklärungen/Begründungen hinter diesen Entscheidungen zu erfassen. Es ist auch nützlich, die Absicht bzw. den Zweck einer bestimmten Lösungsalternative zu erfassen. Im Stand der Technik existiert kein einzelnes Werkzeug zum Erfassen und Verfolgen derartiger wichtiger Informationen einschließlich der Beweggründe und Ziele eines Entwurfs, Fragen, Ideen und Antworten, die während der Untersuchung des Entwurfs gestellt werden, und der gleichen Informationen bezüglich ebenfalls untersuchter Gestaltungsalternativen. Selbst wenn einige der Informationen unter Verwendung eines oder mehrerer unterschiedlicher

Werkzeuge erfaßt wird, weil die Informationen nicht integriert oder einfach zugänglich sind, außer das bestimmte Werkzeug, das zur Erfassung der Informationen benutzt wurde, wird verwendet, werden des weiteren viele der anfänglichen Gestaltungsabsichten und Erklärungen/Beweggründe von Gestaltungsentscheidungen sowie die untersuchten Gestaltungsalternativen üblicherweise nicht effektiv erfaßt oder gehen mit Fortschreiten des Entwicklungszyklus des Produkts verloren.

[0004] Zusätzlich werden im Stand der Technik alle entwurfsbezogenen Informationen, die unter Verwendung eines bestimmten computerbasierten Werkzeugs erfaßt werden, üblicherweise nur über das zur Erzeugung der Daten verwendete Werkzeug gespeichert, verwaltet und abgerufen. Es gibt mehrere Gründe, warum die Möglichkeit, auf Daten, die von einem Werkzeug erzeugt wurden, unter Verwendung anderer Werkzeuge zugreifen zu können, vorteilhaft wäre. Insbesondere können die Informationen, die unter Verwendung eines Werkzeugs erfaßt wurden, für unterschiedliche Menschen von unterschiedlichen Einheiten, die unterschiedliche Aufgaben wahrnehmen, interessant sein. So können beispielsweise Informationen, die während des Entwurfs bzw. der Gestaltung eines Produkts erfaßt wurden, nicht nur für die Gestaltungsingenieure, sondern auch für die Herstellungs- und Testingenieure, Führungskräfte im Produktentwicklungsprozeß, Wartungstechniker, Marketing- und Verkaufspersonal, Disponenten, Personal in der Auftragsverarbeitung, Website-Entwickler und -verwalter, Kunden, Zulieferer usw. nützlich sein.

[0005] Es existiert demnach ein Bedürfnis, Informationen zur Produktentwicklung, nachfolgend als "Innovationsinformationen" bezeichnet, einschließlich Produktideen, Gestaltungsalternativen, Fragen und Antworten, die während des Innovationsprozesses untersucht wurden, Entwurfsentscheidungen usw. zu erfassen, zu speichern und abzurufen. Es besteht auch ein Bedürfnis, die Innovationsinformation in einer werkzeugneutralen Form zu erfassen, die jedem Werkzeug einen Zugriff auf die (und ggf. auch eine Modifikation der) Innovationsinformationen gestattet. Ein derartiges Werkzeug würde die Verfolgung der graduellen Entwicklung vom Entwurf und Entscheidungen über den Entwurf über die Evolution des Produkts gestatten und dabei die funktionalen entwurfsbezogenen Aspekte eines Produkts (anstatt wie im Stand der Technik lediglich die realisierten Konfigurationen des endgültigen Produkts) erfassen und ein Verfolgen und Nachvollziehen deren Entwicklung gestatten. [0006] Demgegenüber wird erfindungsgemäß ein Objektmodell zum Erfassen von Informationen mit den Merkmalen des Anspruchs 1, ein Verfahren zum Erfassen von Informationen mit den Merkmalen des Anspruchs 10 sowie ein Computerprogramm mit den Merkmalen des Anspruchs 18 vorgeschlagen.

[0007] Die vorliegende Erfindung besteht demnach aus einem System und einem Verfahren zum Erfassen und Speichern von und Zugreifen auf Innovationsinformationsdaten in einer beständigen und werkzeugneutralen Form, die einen Datenzugriff durch jedes Werkzeug über eine öffentlich definierte Schnittstelle gestattet. Die Erfindung erfaßt den klaren inkrementalen Aufbau von Innovationsinformationen einschließlich Produktideen, Gestaltungs- bzw. Entwurfsalternativen, Fragen und Antworten, die während des Innovationsprozesses untersucht wurden, Entwurfsentscheidungen usw. Die Innovationsinformationen sind in einer Form dargestellt, die unter Verwendung unterschiedlicher computerbasierter Anwendungen abgefragt und auf verschiedene Weisen wiedergegeben werden kann.

[0008] Die Erfindung erfaßt den evolutionären Aufbau von Informationen, die die Ausgangsidee und die Untersu-

chung eines Produkts über die Zeit betreffen. Während der Untersuchung eines Produkts werden viele Alternativen vorgeschlagen und erforscht. Es werden Entscheidungen gefällt, die Möglichkeiten ausschließen. Mit Fortschreiten der Gestaltung und der Entwicklung ändern sich die gestellten Fragen und zugehörigen Antworten unter dem Einfluß von umfassenden Entscheidungen und dem Aufkommen von mehr ins Detail gehenden Fragestellungen. Teile der Produktdefinition bewegen sich mit unterschiedlicher Geschwindigkeit von der Idee zur vollständigen Definition. Das Innovationsinformationen-Objektmodell der Erfindung unterstützt das Verfolgen bzw. Nachvollziehen der funktionalen gestaltungsorientierten Aspekte des Produkts. Das gestaltungsorientierte Verfolgen stellt somit ein Zeitspektrum vom Untersuchen von Produktideen bis zur vollständigen und für die Produktion freigegebenen Produktdefinition bereit. Die Informationen entwickeln und gestalten sich graduell und werden mit fortschreitendem Gestaltungsprozeß immer detaillierter. Die Möglichkeit für Innovation ist gegeben, solange Fragen offen sind, die kreative Antworten erfordern. Die Auswirkung von Innovation bewegt sich von global zu lokal, wenn Entscheidungen gefällt und Detailfragen angegangen werden.

[0009] Die vorliegende Erfindung umfaßt vorzugsweise ein Objektmodell zur Verwaltung von Innovationsinformationen (nachfolgend kurz als IIM-Objektmodell bezeichnet), das unterschiedliche Objektmodelle erfaßt und speichert, die Informationsartikel und deren zugehörige Beziehungen in einer Objektmodelldatenbank umfassen. Jedes Objektmodell beinhaltet Informationen und Beziehungen, die über eine öffentlich definierte Schnittstelle zugänglich sind. In einer Ausführungsform werden beim Speichern eines Objektmodells in die Objektmodelldatenbank unterschiedliche Typen von Informationsartikeln bzw. Einzelinformationen, die in dem Objektmodell zusammen mit ihren Beziehungen zu anderen Informationsartikeln bzw. Einzelinformationen enthalten sind, in unterschiedlichen relationalen Datenbankdateien, die mit den Informationsartikeln verknüpft sind, gespeichert. Anwendungen, die auf die Objektmodelldatenbank zugreifen, verwenden lediglich die definierte Objektmodellschnittstelle, was zu einer automatischen Trennung und Speicherung von Datentypobjekten in einer beständigen werkzeugneutralen Form führt.

[0010] Die Erfindung erleichtert den Zugriff auf Informationen mittels jeglicher Anwendung über die definierten Objektmodellschnittstellen, unabhängig davon, durch welche Anwendung die Objekte erzeugt wurden. Dadurch entsteht kein "Besitz" an den Daten durch eine Anwendung oder durch das Werkzeug, das die Daten erzeugt hat.

[0011] Die Erfindung erleichtert auch anspruchsvolle Suchen in den Objektmodellen, um interessierende Informationen aus der Gesamtheit der gespeicherten Informationen herauszuziehen. Die Erfindung ist in vielerlei Hinsicht vorteilhaft, einschließlich der Möglichkeit für eine Mehrzahl von Menschen mit unterschiedlichen Aufgaben und Posten, auf die Informationen zuzugreifen und diejenigen Informationen herauszuziehen, die ihrer Aufgabe bzw. ihrer Position entsprechen.

[0012] Weitere Vorteile und Ausgestaltungen der Erfindung ergeben sich aus der Beschreibung und der beiliegenden Zeichnung.

[0013] Es versteht sich, daß die vorstehend genannten und die nachstehend noch zu erläuternden Merkmale nicht nur in der jeweils angegebenen Kombination, sondern auch in anderen Kombinationen oder in Alleinstellung verwendbar sind, ohne den Rahmen der vorliegenden Erfindung zu verlassen.

[0014] Die Erfindung ist anhand von Ausführungsbeispielen

in der Zeichnung schematisch dargestellt und wird im folgenden unter Bezugnahme auf die Zeichnung ausführlich beschrieben.

[0015] Fig. 1 zeigt ein konzeptionelles Objektmodelldiagramm, das die Trennung von Informationsartikeln und ihrer zugeordneten Beziehungen zu anderen Informationsartikeln darstellt.

[0016] Fig. 2 zeigt ein Blockdiagramm eines erfindungsgemäßen Objektmodells zur Verwaltung von Innovationsinformationen.

[0017] Fig. 3 zeigt ein UML-Schnittstellendiagramm (UML: Unified Modelling Language), das eine bevorzugte Ausführungsform der Schnittstelle für das Objektmodell zur Verwaltung von Innovationsinformationen (IIM-Objektmodell) für das in Fig. 2 gezeigte IIM-Objektmodell darstellt.

[0018] Fig. 4 zeigt ein Blockdiagramm der von dem IIM-Objektmodell der Fig. 2 unter Verwendung der in Fig. 3 definierten Schnittstellen erzeugten und unterhaltenen beständigen Speichereinheiten bzw. -entitäten.

[0019] Fig. 5 zeigt ein Anschauungsbeispiel einer relationalen Datenbankdatei, mit der eine erfindungsgemäße Produktidee-Tabelle umgesetzt wird.

[0020] Fig. 6 zeigt ein Anschauungsbeispiel einer relationalen Datenbankdatei, mit der eine erfindungsgemäße Produktanforderungen-Tabelle umgesetzt wird.

[0021] Fig. 7 zeigt ein Anschauungsbeispiel einer relationalen Datenbankdatei, mit der eine erfindungsgemäße Gestaltungsalternativen-Tabelle umgesetzt wird.

[0022] Fig. 8 zeigt ein Anschauungsbeispiel einer relationalen Datenbankdatei, mit der eine erfindungsgemäße Produktfunktion-Tabelle umgesetzt wird.

[0023] Fig. 9 zeigt ein Anschauungsbeispiel einer relationalen Datenbankdatei, mit der eine erfindungsgemäße Anforderungserfüllung-Tabelle umgesetzt wird.

[0024] Fig. 10 zeigt einen Produktidee-Dialog für eine erfindungsgemäß umgesetzte grafische Benutzerschnittstelle.

[0025] Fig. 11 zeigt einen Produktanforderung-Dialog für eine erfindungsgemäß umgesetzte grafische Benutzerschnittstelle.

[0026] Fig. 12 zeigt einen Produktfunktion-Dialog für eine erfindungsgemäß umgesetzte grafische Benutzerschnittstelle.

[0027] Fig. 13 zeigt einen Gestaltungsalternativen-Dialog für eine erfindungsgemäß umgesetzte grafische Benutzerschnittstelle.

[0028] Fig. 1 zeigt ein konzeptionelles Blockdiagramm, das die Trennung von Informationsartikeln bzw. Informationsgegenständen bzw. Einzelinformationen und deren zugehörigen Beziehungen zu anderen Informationsartikeln sowie die Zugreifbarkeit auf die Informationen mittels unterschiedlicher Werkzeuge darstellt. Insbesondere sind eine Ansammlung von Objektmodellen 10a, 10b, 10c, 10d, 10e und 10f, die Informationen und von einer Anzahl unterschiedlicher Werkzeuge 20a, 20b, 20c, ..., 20n während der Entwicklung eines Produktes erzeugte Objektbeziehungen beschreiben, in werkzeugneutraler Form in einem beständigen Speicher 30 gespeichert. Dabei ist es wichtig, daß die Objektmodelle 10a, 10b, 10c, 10d, 10e und 10f nicht einem bestimmten Werkzeug "gehören", insbesondere nicht den Werkzeugen 20a, 20b, 20c, 20n, von denen sie erzeugt wurden. Jedes Objektmodell 10a, 10b, 10c, 10d, 10e und 10f enthält Objekte, die höchst abhängige Objektbeziehungen aufweisen.

[0029] Die Objektmodelle 10a, 10b, 10c, 10d, 10e und 10f verfügen jeweils über eine definierte öffentliche Schnittstelle, die es einem jeden die Schnittstellendefinition verstehenden Werkzeug 20a, 20b, 20c, ..., 20n gestattet, zulässige Daten in den entsprechenden Satz von Objekten zu schreiben.

ben und herauszulesen. Auch wenn die Situation eintreten kann, daß lediglich ein Anwendungswerkzeug ein einzelnes Attribut vollständig versteht (wie ein CAD-Werkzeug, das eine 3D-Geometrie und -topologie versteht), gestattet es die öffentliche Schnittstellendefinition im Grunde genommen jedem Werkzeug, auf Teile des Objekts, das es versteht, einschließlich dessen Beziehungen zu anderen Objekten, zuzugreifen. Beispielsweise erzeugt das CAD-Werkzeug (CAD TOOL) 20a (bspw. SolidDesigner von CoCreate) Daten, die teilweise in dem Objektmodell "CAD-Modell" (CAD MODEL) 10a und teilweise in dem Objektmodell "Produktstruktur" (PRODUCT STRUCTURE) 10b gespeichert werden. Es ist wichtig festzuhalten, daß das CAD-Werkzeug 20a seine interne Datenstruktur oder seine Benutzerschnittstelle nicht ändern muß. Vielmehr muß das CAD-Werkzeug 20a lediglich die Fähigkeit besitzen, nur diejenigen Objekte und Strukturen zu verstehen, die es schreibt und liest, was durch Verwenden einer Erweiterung erreicht werden kann, die eine Import/Exportfähigkeit gestattet; derartige Techniken sind im Stand der Technik gut bekannt. In dem vorliegenden Beispiel greift ein PDM-Werkzeug (PDM TOOL) 20b (PDM: Product Data Management; bspw. WorkManager von CoCreate) auf das Modell "Produktstruktur" 10b und das Modell "Gestaltungsalternative" (DESIGN ALTERNATIVE) 10c zu. Entsprechend muß das PDM-Werkzeug 20b die Fähigkeit besitzen, an dem Modell "Produktstruktur" 10b von dem CAD-Werkzeug 20a vorgenommenen Änderungen handhaben zu können, und entsprechend muß das CAD-Werkzeug 20a die Fähigkeit besitzen, von dem PDM-Werkzeug 20b an dem Modell "Produktstruktur" 10b vorgenommenen Änderungen handhaben zu können. Das gemeinsame Objektmodell (d. h. das Modell "Produktstruktur" 10b), das beide verstehen, unterstützt und verbessert dabei die Kollaboration zwischen dem CAD-Werkzeug 20a und dem PDM-Werkzeug 20b.

[0030] Es ist auch wichtig festzuhalten, daß anderen Werkzeuge (bspw. 20n) auch jederzeit auf die Objektmodelle 10a, 10b, 10c, 10d, 10e und 10f zugreifen können und die Anzahl von Objektmodellen 10a, 10b, 10c, 10d, 10e und 10f jederzeit erweitert werden kann. So erweitert und entwickelt sich die Sammlung von Informationen und Beziehungen mit anderen Objekten entsprechend während des Fortgangs des Produktzyklus, wobei die Entwicklungs- und Gestaltungsaspekte des Produkts festgehalten werden. Zudem gestattet die werkzeugneutrale beständige Form der Objektmodelle sowohl synchrone als auch asynchrone Kollaboration bei der Produktentwicklung, indem vielen unterschiedlichen Benutzern (bspw. Ingenieure, Entscheidungsebene, Verwaltungspersonal und sogar Kunden) mit entsprechender Berechtigung zum Zugriff auf die in den Objektmodellen enthaltenen Daten, die den derzeitigen Stand des Produkts darstellen, gestattet ist.

[0031] Unter den Objektmodellen in der Objektmodelldatenbank 30 befindet sich ein Objektmodell "Verwaltung von Innovationsinformationen" (INNOVATION INFORMATION MANAGEMENT) 10e (kurz: IIM-Objektmodell), das all diejenigen Innovationsinformationen einschließt bzw. zusammenfaßt, die mit der Entwicklung und Gestaltung eines Produkts, einschließlich Produktideen, Gestaltungsalternativen, Fragen und Antworten, die während des Innovationsprozesses untersucht wurden, Gestaltungsentscheidungen usw., verbunden sind sowie ihre Beziehungen zu anderen Objekten in dem IIM-Objektmodell 10e und zu Objekten in anderen Objektmodellen 10a, 10b, 10c, 10d und 10f.

[0032] Objekte in dem IIM-Objektmodell 10e können automatisch von einem oder mehreren der Werkzeuge 20a, 20b, 20c, ..., 20n erzeugt werden, oder sie können konkret

von Nutzern über spezifische IIM-Dialoge (IIM: Innovationsinformationen-Management/Innovation Information Management) erzeugt werden, auf den über die Benutzerschnittstelle der Werkzeuge zugegriffen wird. Zusätzlich kann ein IIM-Werkzeug (TOOL) 20c zur konkreten Eingabe von Innovationsinformationsdaten entwickelt werden. Wie jedoch schon vorstehend angemerkt wurde, "gehören" keinem Werkzeug die Daten in dem IIM-Objektmodell 10e und jedes Werkzeug kann auf die Daten in dem IIM-Objektmodell 10e über öffentlich definierte Schnittstellen, die mit dem IIM-Objektmodell 10e in Verbindung stehen, zugreifen. Diese öffentlich definierten Schnittstellen werden nachfolgend noch ausführlich erläutert.

[0033] Fig. 2 zeigt ein Blockdiagramm, das eine bevorzugte Ausführungsform 100 des IIM-Objektmodells der Fig. 1 darstellt, durch das ein Objektmodell zum Erfassen und Speichern der einzelnen Informationen und deren Struktur, wie sie während der Untersuchungsphase der Entwicklung und Gestaltung eines Produkts in einer werkzeugneutralen beständigen Form entwickelt wurden, bereitstellt. [0034] Wie in Fig. 2 dargestellt umfassen die hauptsächlichsten Informationsgegenstände in dem IIM-Objektmodell 100 die folgenden Objekte: Produktidee (PRODUCT IDEA) 110, Produktanforderung (PRODUCT REQUIREMENT) 120, Gestaltungsalternative (DESIGN ALTERNATIVE) 130, Gestaltungsdarstellung (DESIGN REPRESENTATION) 140, Produktfunktion (PRODUCT FUNCTION) 150, Durchführungsbeschränkung (REGULATORY CONSTRAINT) 160, Gestaltungsabsicht (DESIGN INTENT) 170, Gestaltungsmerkmal bzw. Gestaltungsbedeutung bzw. Gestaltungsanmerkung (DESIGN NOTE) 180, Gestaltungsproblem (DESIGN ISSUE) 190 und Gestaltungsbeschränkung (DESIGN CONSTRAINT) 195.

[0035] Ein Informationsgegenstand (oder Informationsobjekt) "Produktidee" 110 schließt eine Idee betreffend ein Produkt ein. Eine Produktidee kann eine Idee für ein neues Produkt, eine Steigerung oder Verbesserung eines bestehenden Produkts oder die Lösung eines bekannten Problems (wie eine technische Änderung) für ein bestehendes Produkt sein.

[0036] Ein Objekt "Produktanforderung" 120 beinhaltet eine Anforderung, die das Produkt erfüllen muß, soll oder kann. Für ein gegebenes Produkt gibt es üblicherweise viele Anforderungen von vielen verschiedenen Quellen (bspw. Marketing, Kunden, Entwicklung, Herstellung).

[0037] Ein Objekt "Gestaltungsalternative" 130 faßt Informationen zusammen, die eine mögliche Lösung oder Gestaltung für eine Idee darstellen, die in einem Objekt "Produktidee" 110 zusammengefaßt ist. Ein Objekt "Gestaltungsdarstellung" 140 beinhaltet eine Möglichkeit, die in einem Objekt "Gestaltungsalternative" 130 dargestellte vorgeschlagene Lösung oder Gestaltung umsetzen bzw. herstellen.

[0038] Ein Objekt "Produktfunktion" 150 beinhaltet eine Funktion zum Lösen einer Produktanforderung.

[0039] Ein Objekt "Durchführungsbeschränkung" 160 beinhaltet eine Beschränkung, die dem Produkt auferlegt ist und sich außerhalb der Kontrolle der Entwickler/Gestalter befindet. Als Beispiel seien hier gesetzlich festgelegte Kommunikationsfrequenzen im Funkbereich genannt.

[0040] Ein Objekt "Gestaltungsabsicht" 170 beinhaltet eine konkrete Absicht oder Zielsetzung der Gestaltung.

[0041] Ein Objekt "Gestaltungsmerkmal" 180 beinhaltet ein die Gestaltung betreffendes Merkmal.

[0042] Ein Objekt "Gestaltungsproblem" 190 ist mit einer Gestaltungsdarstellung verbunden. Das in einem Gestaltungsproblem-Objekt enthaltene bzw. zusammengefaßte Problem betrifft Bedenken oder eine offene Frage, die nach

Betrachten der Darstellung der Gestaltungsalternative aufgekomen sind.

[0043] Ein Objekt "Gestaltungsbeschränkung" 195 faßt eine Beschränkung zusammen, die der Gestaltungsalternative auferlegt ist und über die Kontrolle besteht.

[0044] Innerhalb des IIM-Objektmodells 100 existieren Beziehungen zwischen den Datenobjekten, wie dies durch die Verbindungslinien zwischen den Objekten dargestellt ist. Z. B. kann jede Produktidee 110 mit verschiedenen Produktanforderungen 120 (wie sie von den Entwicklern und anderen die Gestaltung beeinflussenden Personen definiert werden) verbunden sein, die wiederum keine oder mehrere zugeordnete Produktfunktionsobjekte 150 aufweisen können, die vollständig oder teilweise die in ihrem zugeordneten Produktanforderungsobjekt 120 niedergelegten bzw. zusammengefaßten Anforderungen erfüllen. Jede Produktidee 110 kann auch mit einigen Durchführungsbeschränkungen 160 (wie sie bspw. durch die Industrie oder den Gesetzgeber definiert sein können) verbunden sein, die wiederum durch eine Produktanforderung 120 wie vorstehend beschrieben dargestellt sein können. Jede Produktidee 110 kann auch mit mehreren Gestaltungsalternativobjekten 130 verbunden sein, die wiederum mit Gestaltungsabsichtsobjekten 170 und zugeordneten Produktfunktionsobjekten 150 verbunden sein können. Außerdem kann jede Gestaltungsalternative zugeordnete Gestaltungsbeschränkungen 195 und/oder zugeordnete Gestaltungsdarstellungen 140 aufweisen. Jede Gestaltungsdarstellung 140 kann zugeordnete Gestaltungs-bemerkungen 180 und/oder Gestaltungsprobleme 190 aufweisen.

[0045] Fig. 3 zeigt ein UML-Schnittstellendiagramm, das eine bevorzugte Ausführungsform der IIM-Objektmodell-schnittstelle für das IIM-Objektmodell 100 der Fig. 2 darstellt. Wie in Fig. 3 dargestellt umfassen die für das IIM-Objektmodell 200 der Fig. 2 definierten öffentlichen Schnittstellen folgendes: Produktidee (ProductIdea) 210, Produktanforderung (ProductRequirement) 220, Gestaltungsalternative (DesignAlternative) 230, Gestaltungsdarstellung (DesignRepresentation) 240, Produktfunktion (ProductFunction) 250, Durchführungsbeschränkung (RegulatoryConstraint) 260, Gestaltungsabsicht (DesignIntent) 270, Gestaltungsbemerkung (DesignNote) 280, Gestaltungsproblem (DesignIssue) 290, Gestaltungsbeschränkung (DesignConstraint) 295 und Produktspezifikation (ProductSpecification) 225. Die Attribute für jede Schnittstelle sind wie in Fig. 3 dargestellt. In der IIM-Schnittstellendefinition werden Informationen über jede Produktidee in einem Objekt "Produktidee" 110 (Fig. 2) zusammengefaßt, auf das von der Schnittstelle ProductIdea 210 zugegriffen wird. In dieser Ausgestaltung hat die Schnittstelle ProductIdea 210 keine oder mehrere Schnittstellen RegulatoryConstraint 260, von denen jede mit keiner oder mehreren Schnittstellen ProductRequirement 220 verbunden sein kann. Die Schnittstelle ProductIdea 210 kann auch eine Zuordnung zu keiner oder mehreren Schnittstellen ProductRequirement 220 aufweisen, die aus verschiedenen Quellen stammen können und eine angehängte Beziehung, die über die Schnittstelle RequirementRelationship 222 definiert ist, aufweisen können. Eine Schnittstelle ProductSpecification 225 ist eine Erweiterung der Schnittstelle ProductRequirement 220 und wird dazu verwendet, Spezifikationen für die Anforderungen genauer zu definieren. Jede Schnittstelle ProductRequirement 220 kann auch keiner oder mehreren Schnittstellen ProductFunction 250 zugeordnet sein, die eine Lösung zum teilweisen oder vollständigen Erfüllen einer Produktanforderung bereitstellen. Die Schnittstelle ProductFunction 250 ist in der standardisierten Schnittstelle PDMConfigurationManagement der "Object Management Group (OMG) PDM En-

ablers" definiert, die aus dem Stand der Technik bekannt und in "PDM Enablers: Joint Proposal to the OMG in Response to OMG Manufacturing Domain Task Force RFP 1", Paper mfg/98-02-02 of der Object Management Group (OMG), ausführlich beschrieben ist. Die Schnittstelle PDMConfigurationManagement erweitert den Produktstruktur-Enabler zur Unterstützung von Unternehmen, in denen ein Produkt zum Verkauf in vielen unterschiedlichen Konfigurationen angeboten werden kann. Das Konfigurationsverwaltungsmodul erlaubt eine Spezifikation von Produktklassen und unterscheidenden Produktkonfigurationen. Entsprechend ist das IIM-Objektmodell der Erfindung kompatibel mit den standardisierten PDM-Enabler-Schnittstellen. [0046] Wie ebenfalls dargestellt kann die Schnittstelle ProductIdea 210 keiner oder mehreren Schnittstellen DesignAlternative 230 zugeordnet sein, die einen Zugriff auf keine oder mehrere Objekte "Gestaltungsalternative" 130, die einem bestimmten Objekt "Produktidee" 110 zugeordnet sind, gestatten. Jede Schnittstelle DesignAlternative 230 hat keine oder mehrere Schnittstellen DesignConstraint 295 und keine oder mehrere Schnittstellen DesignRepresentation 240, die jeweils Zugriff auf ein verbundenes Objekt "Gestaltungseinschränkung" 195 und/oder ein zugehöriges Objekt "Gestaltungsdarstellung" 140 gestatten. Jede Schnittstelle DesignRepresentation 240 ist keiner oder mehreren Schnittstellen DesignNote 280 zugeordnet, die Zugriff auf ein Objekt "Gestaltungsbemerkung" 180 gestatten, und mit keiner oder mehreren Schnittstellen DesignIssue 290, die Zugriff auf ein Objekt "Gestaltungsproblem" 190 gestatten.

[0047] Eine Schnittstelle RequirementFulfillment 255 gestattet, auf den Grad der Erfüllung der Produktanforderungen, die die Produktfunktion erfüllt, zuzugreifen. Dies gestattet einem Benutzer, das IIM-Objektmodell später im Hinblick auf diejenigen Gestaltungsalternativen, die bestimmte ggf. prioritätsgewichtete Anforderungen erfüllen, zu untersuchen, um zu ermitteln, welche Gestaltungsalternative einer Erfüllung der Anforderungen mit den höchsten Prioritäten am nächsten kommt.

[0048] Diese Fähigkeit, das System abzufragen, indem wichtige Fragen gestellt werden, und eine automatische Antwort von dem IIM-Objektmodell zu erhalten, ist ein äußerst wichtiger Aspekt, der durch die Erfindung bereitgestellt wird.

[0049] Ein Satz von Entscheidungsschnittstellen gestattet einem Benutzer, unterschiedliche Entscheidungen, die während der Untersuchungsphase des Produkts getroffen wurden, zu erfassen und aufzufinden. Bspw. gestattet eine Schnittstelle ProductRequirementDecision 215 das Auffinden von Fragen, Antworten und daraus resultierenden Entscheidungen, die mit Produktanforderungen verbunden sind. Eine Schnittstelle ProductFunctionDecision 235 gestattet das Auffinden von Fragen, Antworten und daraus resultierenden Entscheidungen, die mit der Verwendung von vorgeschlagenen Produktfunktionslösungen in einer bestimmten Gestaltungsalternative verbunden sind. Eine Schnittstelle DesignAlternativeDecision 275 gestattet das Auffinden von Fragen, Antworten und daraus resultierenden Entscheidungen, die mit der Auswahl von Gestaltungsalternativen verbunden sind. Eine Schnittstelle DesignRepresentationDecision 245 gestattet das Auffinden von Fragen, Antworten und daraus resultierenden Entscheidungen, die mit der Auswahl in der Implementierung einer bestimmten Gestaltungsalternative verbunden sind.

[0050] Die in dem UML-Diagramm der Fig. 3 definierten Schnittstellen sind vorzugsweise in einer objektorientierten Sprache wie C++ oder Java2 implementiert. Die tatsächliche Klassenimplementierung kann von System zu System variieren, da es häufig sinnvoll sein wird, einige der Schnitt-

stellen aus Gründen einer effizienten Implementierung und/oder Leistung (Performance) in einer einzelnen Klasse zusammenzufassen.

[0051] Fig. 4 zeigt ein Blockdiagramm, das eine Ausführungsform der beständigen Speichereinheiten darstellt, die von dem IIM-Objektmodell 100 der Fig. 2 unter Verwendung der in Fig. 3 definierten Schnittstellen erzeugt und unterhalten werden. In dieser Ausführungsform ist mit jeder Schnittstelle eine beständige Speicherdatei, vorzugsweise in Form einer relationalen Datenbank, verbunden. Die von der jeweiligen Schnittstelle eingeschlossenen Daten sind in der jeweiligen zugehörigen relationalen Datenbankdatei gespeichert. Entsprechend gibt es eine separate relationale Datenbankdatei für jede definierte Schnittstelle. In dem dargestellten Ausführungsbeispiel speichert eine Tabelle "Produktidee" (PRODUCT IDEA TABLE) 310 alle verfügbaren Daten, die die Schnittstelle ProductIdea 210 verwenden, eine Tabelle "Produktanforderung" (PRODUCT REQUIREMENT TABLE) 320 speichert alle zugegriffenen Daten, die die Schnittstelle ProductRequirement 220 verwenden, eine Tabelle "Gestaltungsalternativen" (DESIGN ALTERNATIVE TABLE) 330 speichert alle zugegriffenen Daten, die die Schnittstelle DesignAlternative 230 verwenden, eine Tabelle "Gestaltungsdarstellung" (DESIGN REPRESENTATION TABLE) 340 speichert alle zugegriffenen Daten, die die Schnittstelle DesignRepresentation 240 verwenden, eine Tabelle "Produktfunktion" (PRODUCT FUNCTION TABLE) 350 speichert alle zugegriffenen Daten, die die Schnittstelle ProductFunction 250 benutzen, eine Tabelle "Durchführungsbeschränkung" (REGULATORY CONSTRAINT TABLE) 360 speichert alle zugegriffenen Daten, die die Schnittstelle RegulatoryConstraint 260 verwenden, eine Tabelle "Gestaltungsabsicht" (DESIGN INTENT TABLE) 370 speichert alle zugegriffenen Daten, die die Schnittstelle DesignIntent 270 benutzen, eine Tabelle "Gestaltungsbemerkung" (DESIGN NOTE TABLE) 380 speichert alle zugegriffenen Daten, die die Schnittstelle DesignNote 280 benutzen, eine Tabelle "Gestaltungsproblem" (DESIGN ISSUE TABLE) 390 speichert alle zugegriffenen Daten, die die Schnittstelle DesignIssue 290 verwenden. Zusätzlich speichern die Entscheidungstabellen "Produktanforderungsentscheidung" (PRODUCT REQUIREMENT DECISION TABLE) 315, "Produktfunktionsentscheidung" (PRODUCT FUNCTION DECISION TABLE) 335, "Gestaltungsdarstellungsentscheidung" (DESIGN REPRESENTATION DECISION TABLE) 345, "Gestaltungsalternativenentscheidung" (DESIGN ALTERNATIVE DECISION TABLE) 375 jeweils die zugegriffenen Daten, die die Schnittstellen ProductRequirementDecision 215, ProductFunctionDecision 235, DesignRepresentationDecision 245 und DesignAlternativeDecision 275 benutzen. Eine Tabelle "Anforderungserfüllung" (REQUIREMENT FULFILLMENT TABLE) 355 speichert die zugegriffenen Daten, die die Schnittstelle RequirementFulfillment 255 benutzen. Die gestrichelten Linien, die die unterschiedlichen Tabellen miteinander verbinden, stellen einen fremden Schlüssel (foreign key) dar (d. h. eine gemeinsame Spalte in jeder verbundenen relationalen Datenbank), der zur Darstellung der Beziehungen in den Tabellen gespeicherten Daten verwendet wird.

[0052] Fig. 5 ist ein Anschauungsbeispiel einer relationalen Datenbankdatei 410, die eine Tabelle "Produktidee" 310 darstellt. Wie dargestellt, bildet jede Spalte "Beschreibung" (Description) 501 und "Name" 502 auf ein in der Schnittstelle ProductIdea 210 eingeschlossenes bzw. enthaltenes Attribut ab bzw. verweist auf dieses, und jede Reihe bildet auf ein unterschiedliches Objekt "Produktidee" 110 ab.

[0053] Fig. 6 zeigt ein Anschauungsbeispiel einer relationalen

Datenbankdatei 420, die eine Tabelle "Produktanforderung" 320 umsetzt. Wiederum bildet jede Spalte "Produktideenname" (Product Idea Name) 601, "Beschreibung" (Description) 602, "Name" 603, "Kennung" (ID) 604, "Priorität" (Priority) 605 und "Gewichtung" (Weight) 606 jeweils auf ein in der Schnittstelle ProductRequirement 220 eingeschlossenes bzw. enthaltenes Attribut ab bzw. verweist auf dieses, und jede Reihe bildet ein unterschiedliches Objekt "Produktanforderung" 120 ab. In diesem Beispiel ist der Primärschlüssel (d. h. ein einziges Schlüsselfeld (Identifier) für die gesamte Datei) für jedes Objekt "Produktanforderung" 320 dessen Produktanforderungserkennung (in dieser Ausführungsform in Spalte 603 gespeichert). Das Attribut "Produktideenname" des Objekts "Produktidee", mit dem das Objekt "Produktanforderung" 120 verbunden ist, wird als fremder Schlüssel bzw. **Kennbegriff** verwendet. Entsprechend wird eine Spalte "Fremder Schlüssel" 601 bereitgestellt, um jeweilige Objekte "Produktanforderung" auf die zugeordneten Objekte "Produktidee" abzubilden.

[0054] Fig. 7 zeigt ein Anschauungsbeispiel einer relationalen Datenbankdatei 430, die eine Tabelle "Gestaltungsalternative" 330 umsetzt. Die Tabelle umfaßt die Spalten ("Produktideenname" (Product Idea Name) 701, "Beschreibung" (Description) 702, "Name" 703, "Status" 704, "Statusgrund" (Status Reason) 705 und "Geschichte" (History) 706. In diesem Beispiel wird das Attribut "Produktideenname" des Objekts "Produktidee", dem das Objekt "Gestaltungsalternative" 130 zugeordnet ist, als fremder Schlüssel verwendet. Entsprechend wird eine Spalte "Fremder Schlüssel" 701 bereitgestellt, um jeweilige Objekte "Gestaltungsalternative" auf die zugeordneten Objekte "Produktidee" abzubilden.

[0055] Fig. 8 ist ein Anschauungsbeispiel einer relationalen Datenbankdatei 450, die eine Tabelle "Produktfunktion" 350 umsetzt. Die Tabelle umfaßt die Spalten "Produktanforderungskennung" (Product Requirement ID) 801, "Gestaltungsalternativenname" (Design Alternative Name) 802, "Beschreibung" (Description) 803, "Name" 804, "Relevanz" (is relevant for) 805. In diesem Beispiel wird das Attribut "Produktanforderungskennung" des Objekts "Produktanforderung", dem das Objekt "Produktfunktion" 150 zugeordnet ist, als ein fremder Schlüssel und das Attribut "Gestaltungsalternativenname" des Objekts "Gestaltungsalternative", dem das Objekt "Produktfunktion" 150 zugeordnet ist, als anderer fremder Schlüssel verwendet. Entsprechend werden jeweilige Spalten "Fremder Schlüssel" 801 und 802 bereitgestellt, um Objekte "Produktfunktion" 150 auf ihre jeweiligen zugeordneten Objekte "Produktanforderung" 120 und "Gestaltungsalternative" 130 abzubilden.

[0056] Fig. 9 ist ein Anschauungsbeispiel einer relationalen Datenbankdatei 455, die eine Tabelle "Anforderungserfüllung" 355 umsetzt. Die Tabelle umfaßt die Spalten "Produktfunktionsname" (Product Function Name) 901 und "Prozent" (Percent) 902. Wie dargestellt, bildet die Spalte 902 auf das Attribut "Prozent" ab, das über die Schnittstelle RequirementFulfillment 255 eingeschlossen ist, und jede Reihe bildet auf ein anderes Objekt "Produktfunktion" ab. In diesem Beispiel wird das Attribut "Produktfunktionsname" des Objekts "Produktfunktion" 150 als fremder Schlüssel verwendet, entsprechend wird eine Spalte "Fremder Schlüssel" 901 bereitgestellt, um ein Objekt "Anforderungserfüllung" auf sein zugeordnetes Objekt "Produktfunktion" 150 abzubilden.

[0057] Die anderen in Fig. 4 dargestellten Tabellen sind in relationalen Datenbankdateien umgesetzt, die den in den Fig. 5 bis 9 dargestellten ähnlich sind.

[0058] Die Abläufe, durch die bestimmte IIM-Daten erfaßt werden, unterscheiden sich und sind abhängig davon,

welcher Typ von Daten erfaßt wird. Daten können erfaßt werden, wenn ein Benutzer Daten über eine Dialogschnittstelle manuell eingibt (bspw. wenn ein Benutzer eine Produktidee und zugeordnete vorgeschlagene Gestaltungsalternativen, Produktanforderungen und/oder Produktfunktionen unter Verwendung eines Produktideedialogs in der Benutzerschnittstelle der Anwendung eingibt), oder können von einer Anwendung automatisch erzeugt werden (bspw. können Attribute wie Objektkennzeichner (Identifiers), Erzeugungszeit oder Datum der letzten Änderung von der Anwendung zu dem Zeitpunkt, an dem eine Information erfaßt oder geändert wird, automatisch erzeugt oder erfaßt werden). Fig. 10 zeigt beispielhaft einen Produktideedialog "Product Idea" 1000 einer grafischen Benutzerschnittstelle für eine allgemeine Anwendung, die die Fähigkeit hat, auf ein erfindungsgemäßes IIM-Objektmodell 10e zuzugreifen und ein solches zu erzeugen. Wie dargestellt umfaßt der Produktideedialog 1000 Benutzermöglichkeiten zum Eingeben einer Beschreibung "Description" 1002 einer Produktidee und eines Namens "Name" 1004 für die Produktidee. Der Produktideedialog 1000 umfaßt auch Benutzermöglichkeiten (bspw. Knöpfe 1006 und 1008) zum Eingeben vorgeschlagener Produktanforderungen ("Requirement" 1006) und Gestaltungsalternativen ("Design Alternative" 1008) für die Produktidee. Insbesondere wird ein in Fig. 11 dargestellter Produktanforderungs-Dialog 1010 geöffnet bzw. eingeblendet, wenn der Benutzer auf den Knopf "Requirement" 1006 klickt. Wie in Fig. 11 dargestellt gestattet der Produktanforderungsdialog "ProductRequirement" 1010 dem Benutzer, eine Beschreibung ("Description") 1012 der Produktanforderung, einen Namen 1014 für die Produktanforderung und eine der Produktanforderung zugewiesene Priorität ("Priority Level") 1016 einzugeben.

[0059] Auf die Benutzermöglichkeit zum Eingeben einer oder mehrerer vorgeschlagener Produktfunktionen zum teilweisen oder vollständigen Erfüllen der Produktanforderung kann durch Klicken eines Knopfes "Function" 1018 zugegriffen werden, wodurch ein in Fig. 12 dargestellter Dialog "Product Function" 1020 geöffnet bzw. eingeblendet wird. Wie in Fig. 12 dargestellt, kann der Benutzer die Felder "Description" 1022 und "Name" 1024 der Funktion ausfüllen und wahlweise an das Funktionsobjekt einen definierten Abschnitt (der dem System bekannt ist) anhängen, der der Anwendung gestattet, automatisch ein Objekt "Anforderungserfüllung" 255 zu erzeugen, das angibt, wie gut die Produktfunktion die Produktanforderung erfüllt. Unter "Function Definition" 1026 kann eine Funktionsdefinition eingegeben werden.

[0060] Fig. 13 zeigt beispielhaft einen Gestaltungsalternative-Dialog "Design Alternative" 1030, der von dem Dialog "ProductIdea" 1000 durch Klicken auf den Knopf "Gestaltungsalternative" (Design Alternative) 1008 erreicht werden kann. Wie dargestellt, gestattet der Dialog "Design Alternative" 1030 einem Benutzer, eine Beschreibung ("Description") 1032 einen Namen 1034, einen Status 1036 und einen Grund ("Reason") 1038 für den Status 1036 einzugeben. Der Dialog "Design Alternative" 1030 kann auch einen Knopf "Produktfunktion" (Funktion) 1040 umfassen, der den Benutzer zu dem Produktfunktions-Dialog 1020 der Fig. 12 führt, sowie einen Knopf "Produktfunktionsentscheidung" (FunktionDecision) 1042, der einen (nicht dargestellten) Dialog öffnet bzw. einblendet, der dem Benutzer gestattet, eine Entscheidung über jede den Gestaltungsentwurf zugeordnete Produktfunktion einzugeben.

[0061] Wie vorstehend ausführlich beschrieben wurde, stellt die Erfindung einen neuen Weg des Erfassens und Speicherns von sowie des Zugriffs auf Daten im Zusammenhang mit der Verwaltung von Innovationsinformationen dar,

einschließlich dem inkrementalen Aufbau von Innovationsinformationen einschließlich Produktideen, Gestaltungsalternativen, Fragen und Antworten, die während des Innovationsprozesses untersucht wurden, Gestaltungsentscheidungen usw. Die Innovationsinformationen werden in einer Form dargestellt, auf die in unterschiedlicher Weise zugegriffen werden kann und die auf unterschiedliche Weise darstellbar ist, unter Verwendung von computerbasierten Anwendungen.

[0062] Die Erfindung erfaßt den evolutionären Aufbau von Informationen, die die ursprüngliche Idee (Ausgangsidee) und die Untersuchung eines Produkts über die Zeit betreffen. Während der Untersuchung eines Produkts werden viele Alternativen vorgeschlagen und erforscht. Entscheidungen werden gefällt, die Möglichkeiten ausräumen bzw. ausschließen. Mit fortschreitender Gestaltung ändern sich die gestellten Fragen und zugehörigen Antworten, da umfassende Entscheidungen gefällt werden und detailliertere Fragestellungen auftreten. Teile der Produktdefinition bewegen sich von dem Status der Idee zu der fertigen Definition mit unterschiedlicher Geschwindigkeit. Das erfindungsgemäße Objektmodell zur Verwaltung von Innovationsinformationen unterstützt dabei das Auffinden bzw. Nachvollziehen der funktionalen Wie-Ausgeführt-Aspekte des Produkts. Dieses Auffinden stellt somit ein Zeitspektrum von der Untersuchung von Produktideen zu der vollständigen und für die Produktion freigegebenen Produktdefinition bereit. Die Informationen entwickeln sich dabei graduell und werden mit fortschreitendem Gestaltungsprozeß detaillierter. Die Möglichkeit für Innovation ist so lang gegeben, so lange Fragen offen bleiben, die kreative Antworten benötigen. Die Auswirkung von Innovation bewegt sich von der umfassenden Ebene auf detailliertere Ebenen, wenn Entscheidungen gefällt werden und Details berücksichtigt werden.

[0063] Die Erfindung vereinfacht den Zugriff auf Informationen durch jede Anwendung über die definierten Objektmodellschnittstellen, unabhängig davon, welche Anwendung die Objekte erzeugt hat. Somit "gehören" die Daten keiner Anwendung und auch keinem Werkzeug, das die Daten erzeugt hat.

[0064] Die Erfindung vereinfacht auch anspruchsvolle Suchen in den Objektmodellen, um interessierende Informationen aus der Gesamtheit der gespeicherten Informationen herausfinden. Die Erfindung ist aus vielen Gründen vorteilhaft, einschließlich der Tatsache, daß viele Benutzer mit unterschiedlichen Aufgaben auf eine Art und Weise, die ihrer jeweiligen Aufgabe bzw. Bedeutung gerecht wird, auf die Informationen zuzugreifen und Informationen herauszuziehen.

[0065] Wie nach der Lektüre der vorstehenden ausführliche Beschreibung offensichtlich wird, verfügt die Erfindung über einige Vorteile gegenüber dem Stand der Technik. Das erfindungsgemäße IIM-Objektmodell erhöht die Effizienz beim Nachvollziehen und Verstehen, wie und warum eine Produktgestaltung eine derzeitige Konfiguration aufweist. Das erfindungsgemäße IIM-Objektmodell gestattet das Erfassen und Nachvollziehen von funktionalen Wie-Gestalt-Aspekten des Produkts anstelle der Wie-Ausgeführt-Konfiguration des endgültigen Produkts und stellt dadurch ein Zeitspektrum von der anfänglichen Untersuchung von Produktideen bis zu der vollständigen und zur Produktion freigegebenen Produktdefinition bereit. Die Informationen entwickeln sich graduell und werden mit fortschreitendem Gestaltungsprozeß immer detaillierter.

1. Objektmodell zum Erfassen von Informationen, die produktinnovationsbezogene Daten betreffen, mit einer Produktideeschnittstelle zum Erfassen einer Idee für ein Produkt in einem Produktideeobjekt und einer Gestaltungsalternativenschnittstelle zum Erfassen einer Vielzahl von Gestaltungsalternativen für das Produkt in einer Vielzahl von jeweiligen Gestaltungsalternative-Objekten.
2. Objektmodell nach Anspruch 1, mit einer Produktanforderungsschnittstelle zum Erfassen einer Anforderung für die Produktidee in einem Produktanforderungsobjekt.
3. Objektmodell nach Anspruch 2, mit einer Produktfunktionsschnittstelle zum Erfassen einer Funktion zum Erfüllen der Produktanforderung in einem Produktfunktionsobjekt.
4. Objektmodell nach Anspruch 3, mit einer Produkt-erfüllungsschnittstelle, die erfährt, wie gut die Produktfunktion die Produktanforderung erfüllt.
5. Objektmodell nach Anspruch 1, mit einer Gestaltungsdarstellungsschnittstelle zum Erfassen einer Darstellung der Gestaltungsalternative in einem Gestaltungsdarstellungsobjekt.
6. Objektmodell nach Anspruch 1, mit einer Entscheidungsschnittstelle zum Erfassen einer Entscheidung in einem Produktanforderungsobjekt, wobei diese Entscheidung entweder die Produktidee oder die Gestaltungsalternative betrifft.
7. Objektmodell nach Anspruch 1, bei dem Produktideeobjekt und jedes Gestaltungsalternative-Objekt in einer werkzeugneutralen beständigen Form gespeichert werden.
8. Objektmodell nach Anspruch 2, bei dem Produktideeobjekt, jedes Gestaltungsalternative-Objekt und jedes Produktanforderungsobjekt in einer werkzeugneutralen beständigen Form gespeichert werden.
9. Objektmodell nach Anspruch 5, bei dem Produktideeobjekt, jedes Gestaltungsalternative-Objekt und jedes Gestaltungsdarstellungsobjekt in einer werkzeugneutralen beständigen Form gespeichert werden.
10. Verfahren zum Erfassen von Informationen, die produktinnovationsbezogene Daten betreffen, mit den folgenden Schritten:
Erfassen einer Idee für ein Produkt in einem Produktideeobjekt, und
Erfassen einer Vielzahl von Gestaltungsalternativen für das Produkt in einer Vielzahl von entsprechenden Gestaltungsalternative-Objekten.
11. Verfahren nach Anspruch 10, mit dem Schritt des Erfassens einer Anforderung für die Produktidee in einem Produktanforderungsobjekt.
12. Verfahren nach Anspruch 11, mit dem Schritt des Erfassens einer Funktion zum Erfüllen der Produktanforderung in einem Produktfunktionsobjekt.
13. Verfahren nach Anspruch 10, mit dem Schritt des Erfassens einer Darstellung der Gestaltungsalternative in einem Gestaltungsdarstellungsobjekt.
14. Verfahren nach Anspruch 10, mit dem Schritt des Erfassens einer Entscheidung in einem Entscheidungsobjekt, wobei die Entscheidung entweder die Produktidee oder die Gestaltungsalternative betrifft.
15. Verfahren nach Anspruch 1, mit dem Schritt des Speicherns des Produktideeobjekts und jedes Gestaltungsalternative-Objekts in einer werkzeugneutralen beständigen Form.
16. Verfahren nach Anspruch 10, mit dem Schritt des

Speicherns des Produktideeobjekts, jedes Gestaltungsalternative-Objekts und des Produktanforderungsobjekts in einer werkzeugneutralen beständigen Form.

17. Verfahren nach Anspruch 13, mit dem Schritt des Speicherns des Produktideeobjekts, jedes Gestaltungsalternative-Objekts und jedes Gestaltungsdarstellungsobjekts in einer werkzeugneutralen beständigen Form.

18. Computerprogramm mit Programmanweisungen, die ein Verfahren zum Erfassen von Informationen, die produktinnovationsbezogene Daten betreffen, umsetzen, wobei das Verfahren die folgenden Schritte umfaßt:

Erfassen einer Idee für ein Produkt in einem Produktideeobjekt, und

Erfassen einer Vielzahl von Gestaltungsalternativen für das Produkt in einer Vielzahl von jeweiligen Gestaltungsalternative-Objekten.

19. Computerprogramm nach Anspruch 18, bei dem das Verfahren den Schritt des Speicherns des Produktideeobjekts und jedes Gestaltungsalternative-Objekts in einer werkzeugneutralen beständigen Form umfaßt.

20. Computerprogramm nach Anspruch 18, wobei das Verfahren die folgenden Schritte umfaßt:

Erfassen einer Anforderung für die Produktidee in einem Produktanforderungsobjekt,

Erfassen einer Darstellung der Gestaltungsalternative in einem Gestaltungsdarstellungsobjekt, und

Speichern des Produktideeobjekts, jedes Gestaltungsalternative-Objekts, des Produktanforderungsobjekts und jedes Gestaltungsdarstellungsobjekts in einer separaten relationalen Datenbank, wobei Verknüpfungen zwischen den einzelnen Objekten unter Verwendung fremder Schlüssel erfaßt werden.

Hierzu 8 Seite(n) Zeichnungen

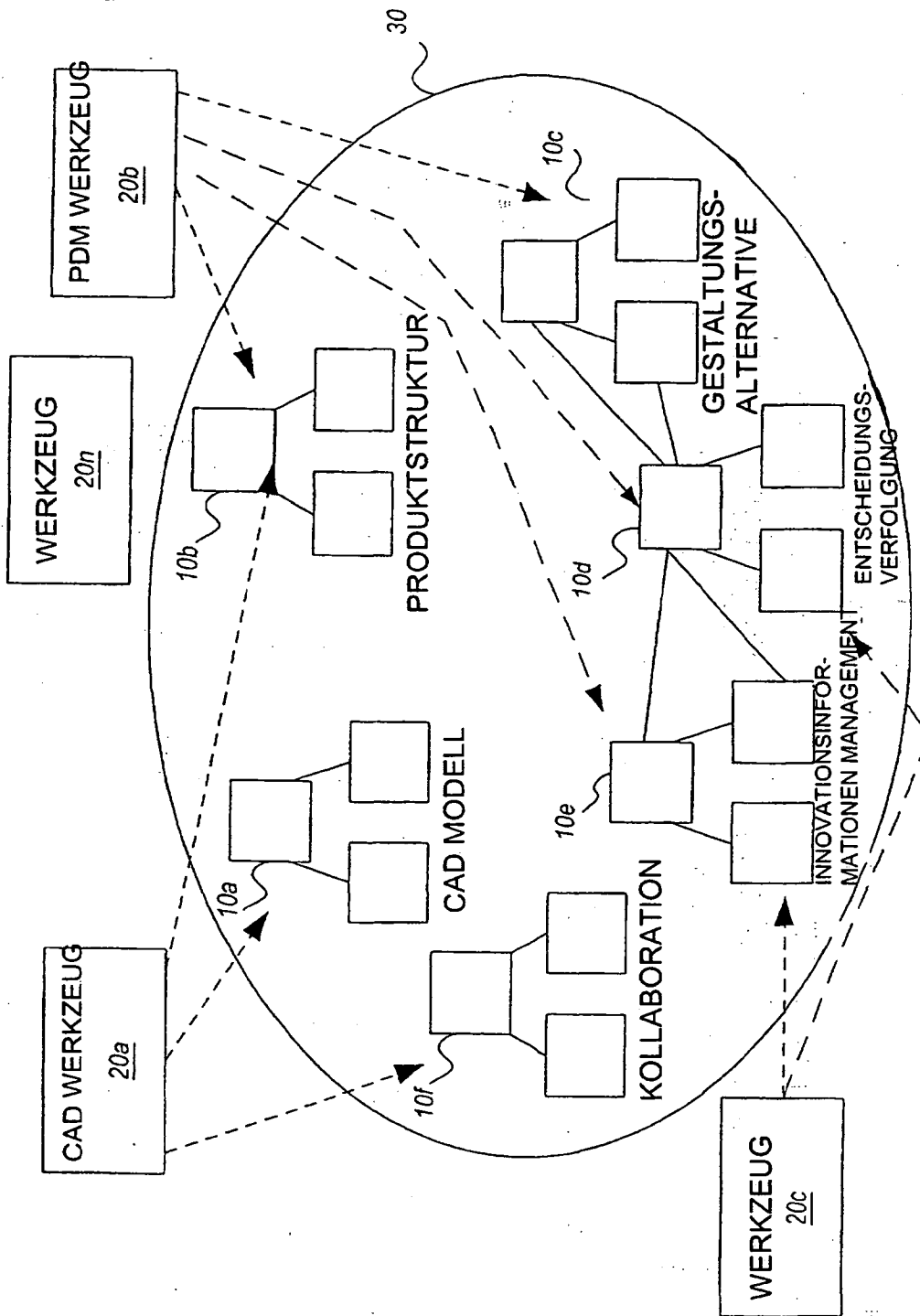


FIG. 1

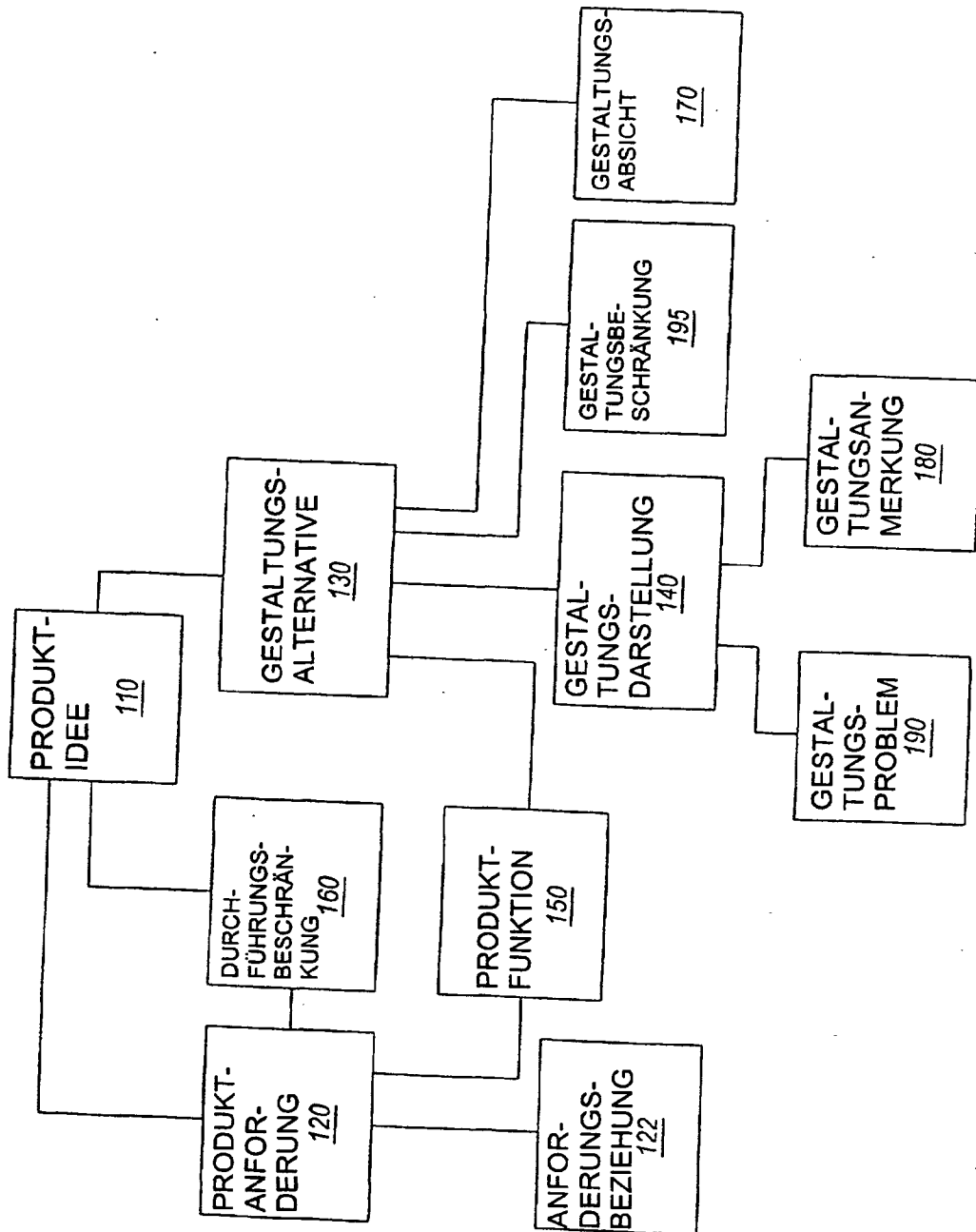


FIG. 2

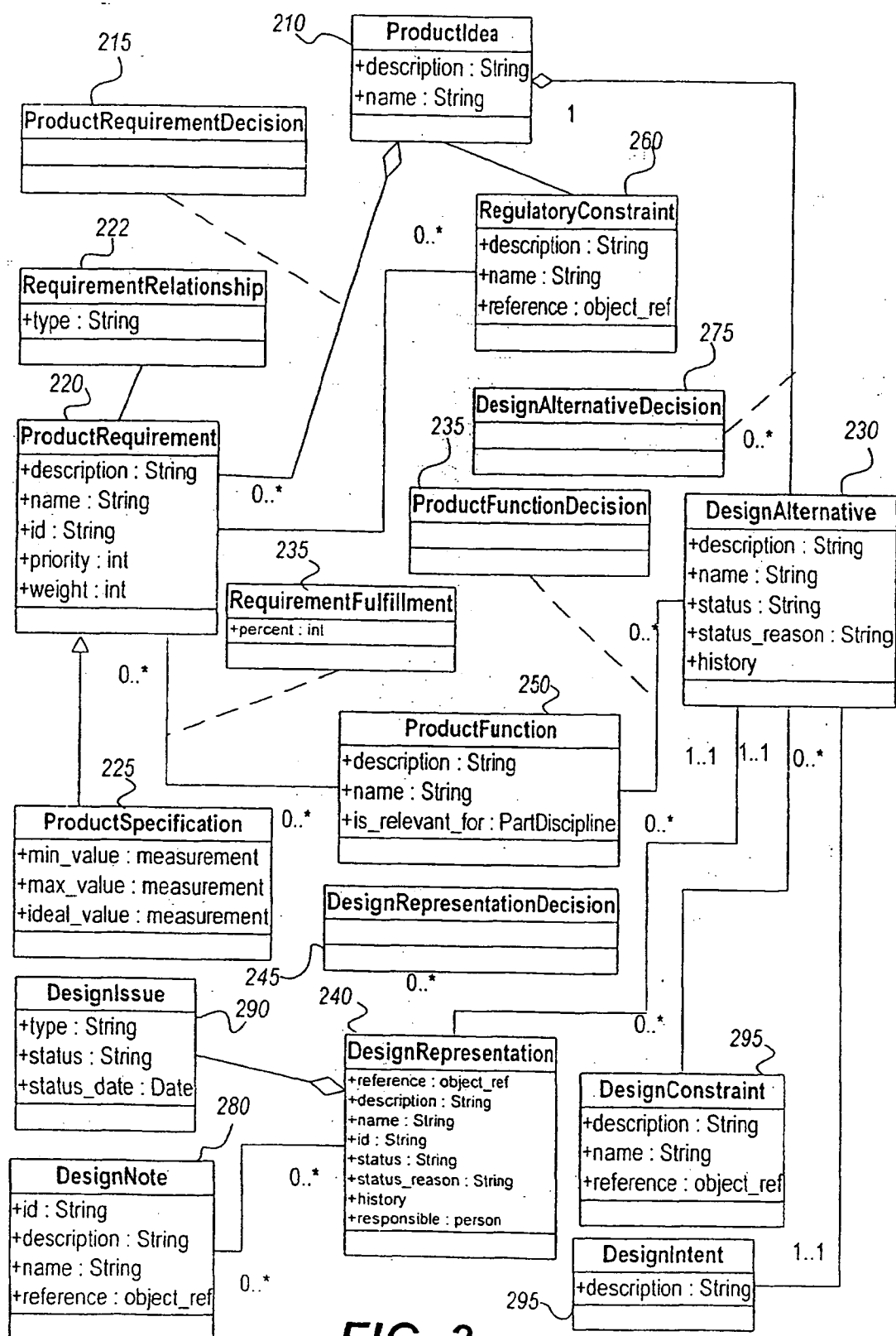


FIG. 3

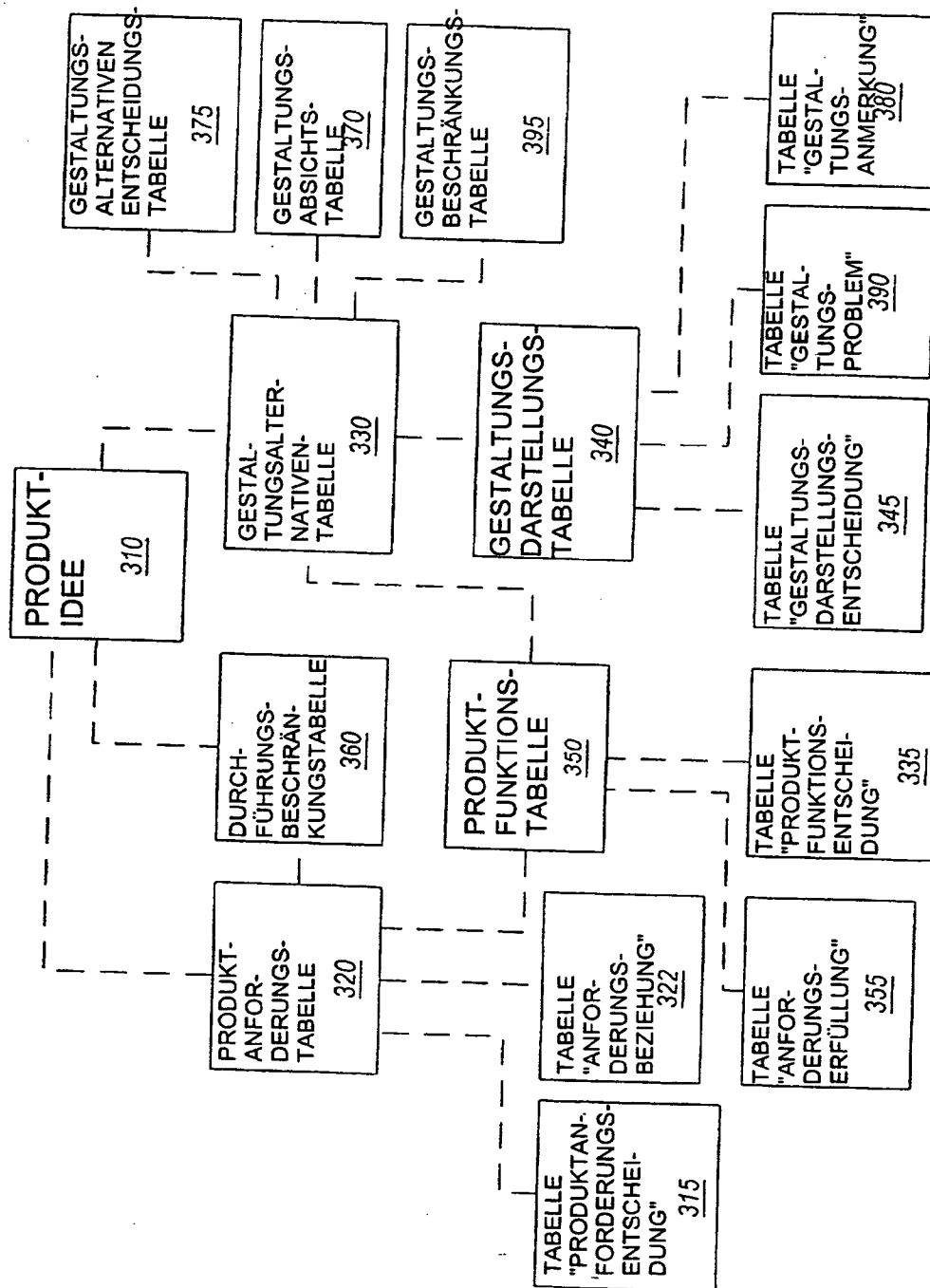


FIG. 4

410

BESCHREIBUNG	NAME

FIG. 5

420

PRODUKTIDEE NAME	BESCHREIBUNG	NAME	KENNUNG	PRIORITÄT	GEWICHT

FIG. 6

430

PRODUKTIDEE NAME	BESCHREIBUNG	NAME	STATUS	STATUS-GRUND	GESCHICHTE

FIG. 7

450 ANFORDERUNGS- KENNUNG	801	802 PRODUKT- ALTERNATIVEN- NAME	803 GESTÄLTUNGS- BESCHREIBUNG	804 NAME	805 RELEVANZ	806

FIG. 8

455 PRODUKT- FUNKTIONS- NAME	901	902 PROZENT

FIG. 9

FIG. 10 is a dialog box titled "Product Idea". It features a "NAME:" label (1004) followed by a text input field. Below this is a "Description" label (1002) followed by a large text area. At the bottom, there are three buttons: "Requirement" (1006), "Design Alternative" (1008), and "Cancel".

FIG. 10

FIG. 11 is a dialog box titled "Product Requirement". It features a "NAME:" label (1014) followed by a text input field. Below this is a "Description" label (1012) followed by a large text area. At the bottom, there is a "PRIORITY LEVEL:" label (1016) followed by a dropdown menu showing the value "1". To the right of the dropdown are two buttons: "Function" (1018) and "Cancel".

FIG. 11

FIG. 12 is a dialog box titled "Product Function". It features a "NAME:" label (1024) followed by a text input field. Below this is a "Description" label (1022) followed by a large text area. At the bottom, there is a "Function Definition:" label (1026) followed by a dropdown menu. To the right of the dropdown is a "Cancel" button.

FIG. 12

1030

Design Alternative

1034 NAME:

Description

1032

1036 STATUS:

STATUS REASON:

Function

Function Decision

Cancel

1040 1042 1038

FIG. 13